URBANITE

Supporting the decision-making in urban transformation with the use of disruptive technologies

Deliverable D4.3

URBANITE Policy Decision Model

Editor(s):	Erik Dovgan, Maj Smerkol, Miljana Shulajkovska, Gjorgji Noveski
Responsible Partner:	Jožef Stefan Institute
Status-Version:	Final - 1.0
Date:	07. 04. 2022
Distribution level (CO, PU):	Public

Project Number:	GA 870338
Project Title:	URBANITE

Title of Deliverable:	URBANITE policy decision model
Due Date of Delivery to the EC:	31/03/2022

Workpackage responsible for the Deliverable:	WP4 – Algorithms and simulation techniques for decision-makers
Editor(s):	Jožef Stefan Institute
Contributor(s):	JSI
Reviewer(s):	Nerea Rojas (MLC/BILBAO)
Approved by:	All Partners
Recommended/mandatory readers:	WP3, WP5, WP6

Abstract:	This document presents the URBANITE decision support system with policy decision models, adapted for the needs of each use case. This deliverable is the result of Task T4.2.
Keyword List:	Decision support system, recommender engine
Licensing information:	
Disclaimer	

Document Description

Document Revision History

Version	Date	Modifications Introduced	
		Modification Reason	Modified by
v0.1	03/03/2022	Draft ToC	JSI
v0.2	07/03/2022	Multi-attribute and multi-objective decision modelling, initial version of DSS	JSI
v0.3	08/03/2022	Upgrade of various sections	JSI
v0.4	10/03/2022	Draft text finished, formatting	JSI
v0.5	16/03/2022	Added policy proposal methods	JSI
v0.6	18/03/2022	Submitted for internal review	JSI
v0.7	24/03/2022	Reviewed and sent to editor	MLC
V0.8	31/03/2022	Final edits	JSI
V1.0	07/04/2022	Final version	TEC

Table of Contents

List of Figures

Terms and abbreviations

EC	European Commission
DSS	Decision Support System
KPI	Key Performance Indicator
DEXi	A Program for Multi-Attribute Decision Making
HBEFA	Handbook on Emission Factors for Road Transport
ML	Machine Learning
EA	Evolutionary Algorithm
GUI	Graphical User Interface

Executive Summary

This document is the third deliverable of WP4, the work package focused on the development of advanced AI algorithms for analysis of big data and simulation techniques, as support for decision-makers to tackle complex policy problems. This deliverable presents the decision support system (DSS) that is being developed for the URBANITE project.

The document has five main blocks devoted to: multi-attribute and multi-objective decision modelling, overview of the DSS approach, initial version of DSS, city-specific DSS models, and policy proposal. The document begins with a description of the main approaches and methods for multi-attribute and multi-objective decision modelling, which are relevant for the URBANITE DSS. Next, it describes the DSS approach and DSS interaction with other URBANITE components. It continues with the description of the initial version of DSS, which was developed based on the initial description of the use cases. The DSS is then redefined based on the updated use cases that require city-specific DSS models. Finally, it describes the approach for policy effect modelling and policy proposal.

This document constitutes an intermediate step of the development of the URBANITE DSS, since the final implementation of DSS including the recommendation system will be available and described in Month 30 (Deliverable D4.6). Therefore, the policy evaluation module including the decision models has already been developed, while the recommendation system is still under development and will be concluded in the next 6 months.

Although use case requirements were already defined, some modifications may be required. Therefore, the developed policy evaluation module might still be upgraded. However, we do not foresee the need for significant modifications, since the module is very flexible, allowing redefinition of the decision models on the fly. Nevertheless, if new KPIs are defined, the data preprocessing module will need to be upgraded. On the other hand, the recommendation system is still under development thus new requirements can still be considered.

1 Introduction

Deliverable D4.3 presents the overview of the URBANITE Decision Support System (DSS). It includes the initial as well as the upgraded version of DSS, and the module for policy proposal.

This document is part of work package WP4 “Algorithms and simulation techniques for decision-makers” and is the outcome of Task T4.2 “Recommendation engine and policy support systems”.

It presents the approach for multi-attribute and multi-objective decision modelling, overview of the DSS approach, initial version of DSS, city-specific DSS models, and the policy proposal module.

1.1 About this deliverable

The deliverable describes the main approaches and methods for multi-attribute and multi-objective decision modelling, which are relevant for the URBANITE DSS. Based on these approaches, the URBANITE DSS is given, including the description of the interaction between DSS and other URBANITE components. The DSS has been developed with an incremental approach. First, the initial version of DSS is described, and second, an enhanced city-specific DSS is presented. Finally, the approach for policy effect modelling and policy proposal is given.

This deliverable is an intermediate deliverable of Task T4.2. Thus, the final implementation of the recommendation system will be available in Month 30 and described in Deliverable D4.6. Therefore, this deliverable provides the current version of the recommendation system, which will be further developed in the future.

1.2 Document structure

This document is organised into eight main blocks:

- Introduction explains the rationale of this document and the structure in more detail.
- The second block (section 2) describes the main approaches and methods for multi-attribute and multi-objective decision modelling, which are relevant for the URBANITE DSS.
- The third block presents the DSS approach and DSS interaction with other URBANITE components (section 3.1).
- The fourth block describes the initial version of DSS (section 3.2), which was developed based on the initial description of the use cases.
- The fifth block presents a redefined approach based on the updated use cases that require city-specific DSS models (section 3.3).
- The sixth block describes the approach to policy effect modelling and policy proposal (section 3.4).
- The seventh block identify the installation instructions, the user manual location, licensing information and the repository URL for downloading the source code of the tool and also the specific files and parameters that define the decision model for the URBANITE use cases (section 4).
- Finally, the conclusion summarises the key points of the document and outlines the future work.

2 Multi-Attribute and Multi-Objective Decision Modelling

The task of the URBANITE DSS is to evaluate, search for, and propose the best mobility policies for the smart cities. To this end, it implements several concepts from the decision-making approaches, such as multi-attribute decision modelling and multi-objective decision making. In addition, it is based on the established DEXi decision modelling approach. This section provides an overview of the theoretical background of the developed DSS.

2.1 Theoretical background

Real-life problems typically involve more than one objective function to be optimised simultaneously and are called multi-objective problems. In contrast to the single-objective problems, the result of multi-objective problems consists of more than one solution, each with different trade-offs between objectives. These solutions cannot be found by optimising one objective at a time with single-objective approaches but require simultaneous optimization of all objectives.

A multi-objective problem can be formulated as a set of objective functions that need to be optimised, either minimised or maximised. In addition, a set of constraints can be defined, which define the feasibility of the solutions. A mathematical definition of the multi-objective problem is:

$$
\begin{array}{lll}
\text{Minimise/maximise} & f_m(x), & m = 1, 2, \ldots, M_{mo}; \\
\text{subject to} & g_j(x) \geq 0, & j = 1, 2, \ldots, J_{mo}; \\
 & h_k(x) = 0, & k = 1, 2, \ldots, K_{mo}; \\
 & x_i^{(L)} \leq x_i \leq x_i^{(U)}, & i = 1, 2, \ldots, n_{mo}.
\end{array}
$$

Explanation:

- x is a solution that is encoded as a vector of decision variables $x = (x_1, x_2, \ldots, x_n)^T$
- x_i is a decision variable. In our case, decision variables are Key Performance Indicators (KPIs), city attributes or similar
- $x_i^{(L)}$ and $x_i^{(U)}$ are lower and upper bounds of the decision variable x_i
- D_{mo} is the decision (variable) space, defined by the lower and upper bounds
- $f_m(x)$ are objective functions
- $g_j(x)$ and $h_k(x)$ are inequality and equality constraints

A solution that satisfies all the constraints and variable bounds is a feasible solution; otherwise, it is an infeasible solution. The set of feasible solutions is called the feasible region S_{mo}.

In addition to the decision space, there is the objective space Z_{mo}, where each solution x is represented with a point, denoted by $f_m(x) = z = (z_1, z_2, \ldots, z_{M_{mo}})^T$, as shown in Figure 1.

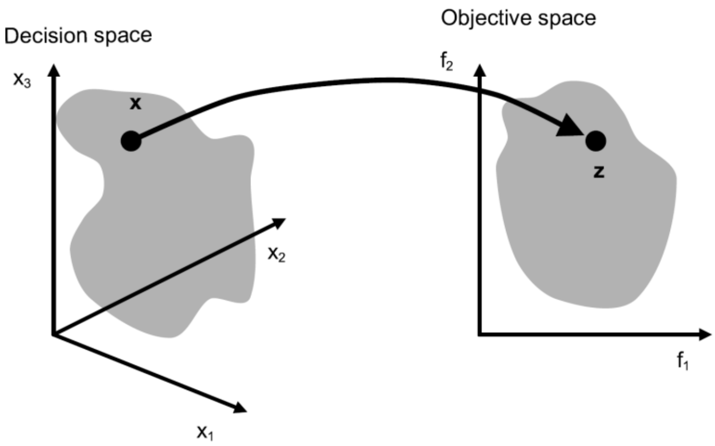

Figure 1 Decision space and objective space.

Linearity and convexity

The multi-objective problem is linear if all objective and constraint functions are linear; otherwise, the resulting problem is a nonlinear multi-objective problem. Most real-world problems are nonlinear.

A multi-objective problem is convex if all objective functions are convex and the feasible region is convex, or if all inequality constraints are nonconvex and equality constraints are linear. If the function is convex, a local minimum is a global minimum [1].

Solution dominance and nondominated solutions

A key relation in the multi-objective problems is the dominance relation. A solution $x^{(1)}$ dominates the other solution $x^{(2)}$, i.e., $x^{(1)} \preceq x^{(2)}$, if both conditions 1 and 2 are true [1]:

1. The solution $x^{(1)}$ is no worse than $x^{(2)}$ in all objectives.
2. The solution $x^{(1)}$ is strictly better than $x^{(2)}$ in at least one objective.

With the dominance relation we can compare solutions with multiple objectives. It also enables us to define and find nondominated solutions, i.e., solutions that are not dominated by any other solution. When solving the multi-objective problems, we are searching for nondominated solutions. On the other hand, the dominated solutions can be disregarded since they are not interesting for the user due to the fact that each such solution is dominated by at least one nondominated solution; that is, it is worse than at least one nondominated solution.

The dominance relation is illustrated in Figure 2.

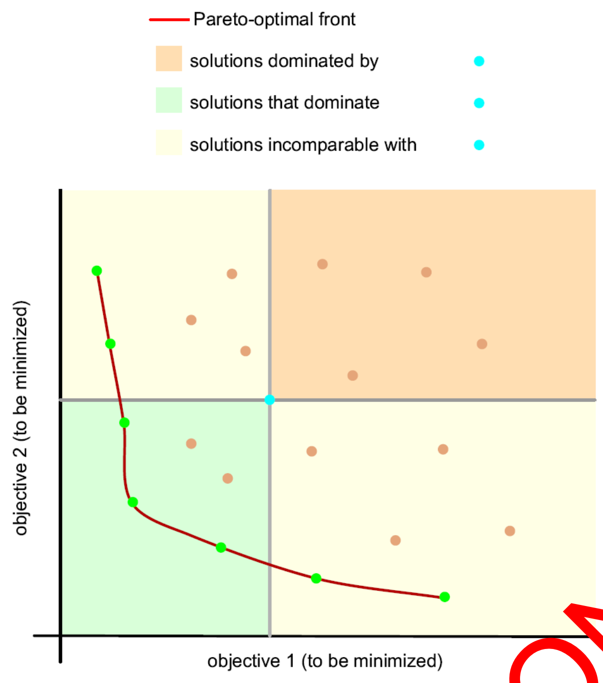

Figure 2 The dominance relation

Dominance relation and its properties

The dominance relation defines three outcomes of the comparison of two solutions x_1 and x_2:

(a) solution x_1 dominates solution x_2, or
(b) solution x_1 is dominated by solution x_2, or
(c) solutions x_1 and x_2 do not dominate each other and are therefore incomparable.

The dominance relation is [1]:

- not reflexive, since a solution does not dominate itself,
- not symmetric,
- asymmetric, since if solution x_1 dominates solution x_2, then solution x_2 does not dominate solution x_1,
- not antisymmetric,
- transitive, and
- a strict partial order relation.

In addition, if solution x_1 does not dominate solution x_2, this does not imply that solution $x_2$dominates solution x_1.

Pareto-optimal solutions

The discovery of nondominated solutions can be incremental, so not all such solutions might be discovered at once. However, when all feasible solutions are considered/evaluated, the set of nondominated solutions is the final one, thus no additional nondominated solutions can be found. In this case, these nondominated solutions are the Pareto-optimal solutions forming the Pareto-optimal front (see Figure 2). Therefore, the task of multi-objective decision making is to find Pareto-optimal solutions, i.e., solutions that belong to a set of nondominated solutions that has the following properties:

- Each solution from the set is not dominated by any other solution from the set.
- Each solution that does not belong to the set is dominated by at least one solution from the set.

Although the goal is to find Pareto-optimal solutions, in many cases this is not feasible, e.g., due to an infinite number of solutions or due to nonexistence of an exact mathematical procedure to solve the multi-objective problem. In these cases, an adapted goal is to find nondominated solutions.

Multi-objective problem reduction to single-objective problem

A multi-objective problem has conflicting objectives. Two objectives are conflicting if one objective worsens when the other one improves. As a result, there are several Pareto-optimal solutions, each with a different trade-off between objectives. On the contrary, if the objectives are not conflicting, there is only one Pareto-optimal solution and this solution can be found by optimising one of the objectives. In this case, the problem is reduced to a single-objective problem.

Boundary solutions of the multi-objective problem

Multi-objective problems have the following special solutions (see Figure 3):

- Ideal objective vector
- Utopian objective vector
- Nadir objective vector

The ideal objective vector consists of optimal values for each objective function. Note that in the majority of cases, the ideal vector corresponds to a non-existent solution. This is not true only when the objectives are not conflicting to each other. In such a case, only one optimal solution exists. Although the ideal vector in most cases corresponds to a hypothetical solution, it serves as a reference point, since solutions closer to the ideal vector are better. Moreover, this vector can be used to normalise objective values in a common range.

The utopian objective vector is strictly better than (and not equal to) the ideal objective vector in each component.

The nadir objective vector consists of the worst values for each objective function in the entire Pareto-optimal set. Ideal and nadir objective vectors can be used to normalise the objective values in the Pareto-optimal region.

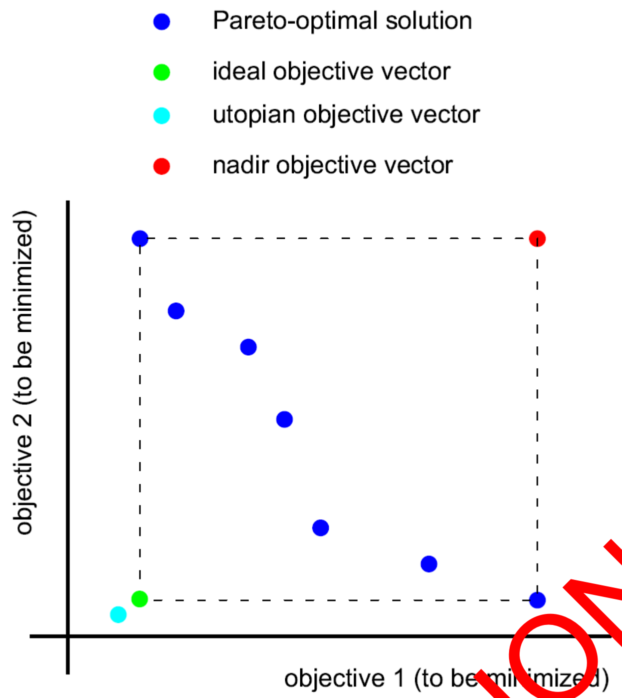

Figure 3 Boundary solutions of multi-objective problems.

2.2 DEXi decision modelling

DEXi defines the approach of multi-attribute decision making. It consists of qualitative multi-attribute decision models that are built with an interactive process. It supports complex decision-making tasks where several objectives are optimised by the decision-makers [2]. An example of multi-attribute decision models is shown in Figure 4.

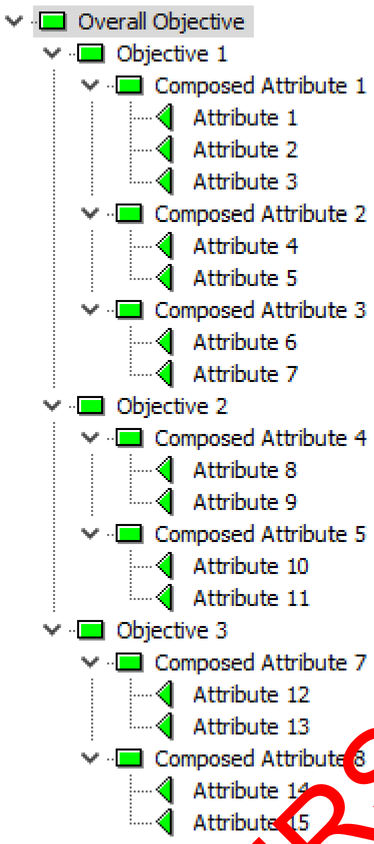

Figure 4 An example of multi-attribute decision models.

The multi-attribute decision models are hierarchical structures that represent the decomposition of the decision problems into subproblems. For each subproblem, i.e., each internal node, an aggregation function called utility function needs to be defined. Such subproblems are smaller, less complex, and therefore easier to solve compared to the complete problem. This is particularly suitable for decision-makers, since they do not need to define an utility function that qualitatively aggregates the attributes for the entire problem at once, but are able to define several simpler utility functions, one for each (simpler) subproblem independently. An example of the utility function for a subproblem is shown in Figure 5.

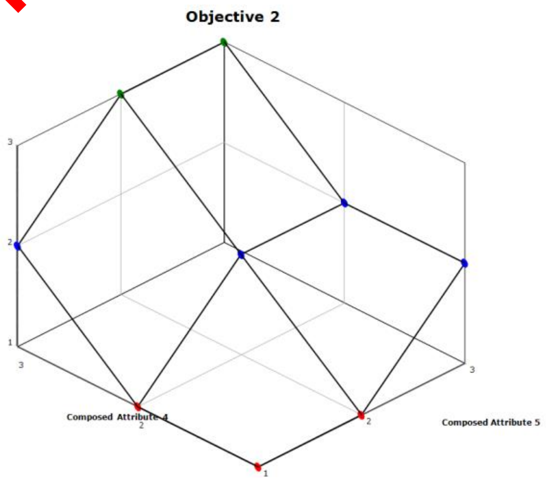

Figure 5 An example of the aggregation, i.e, utility function for a subproblem within the entire decision model shown in Figure 4.

DEXi enables the development of qualitative models that consist of qualitative, i.e., discrete attributes and objectives. Consequently, DEXi is best-suited for classification decision-analysis tasks with a finite number of predefined categories. In addition, DEXi supports multi-objective decision making with the dynamic selection of the objectives relevant for the decision-makers. The discrete objectives and the multi-objective space of sample solutions are shown in Figure 6.

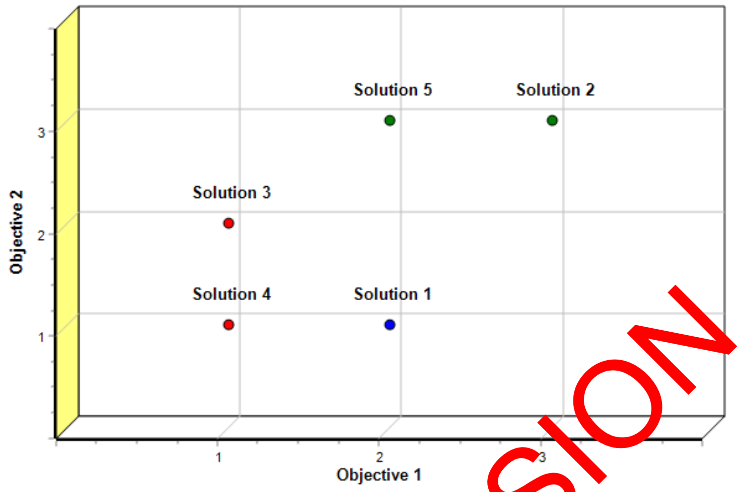

Figure 6 An example of the discrete objectives and the multi-objective space of sample solutions.

3 URBANITE Decision Support System

This section describes the URBANITE Decision Support System (DSS) for multi-objective decision making, which is based on the DEXi decision modelling. Its initial implementation is described in [3].

The goal of the use of DSS is to define the best mobility policies. A mobility policy consists of a set of actions to be applied within the urban area (such as closing a specific road for cars) and can be proposed either by decision-makers or by policy proposal methods. Both take into account the policy evaluation, computed by DSS. The main difference is that decision-makers rely on expert knowledge and define the mobility policies by hand, while policy proposal methods apply pattern-recognition approaches, process a possibly huge amount of data, and propose mobility policies automatically.

The overall URBANITE system consists of a set of modules, e.g., tools for the involvement of various types of stakeholders (general public, etc.). However, from the point of view of DSS, only the modules presented in Figure 7 are relevant.

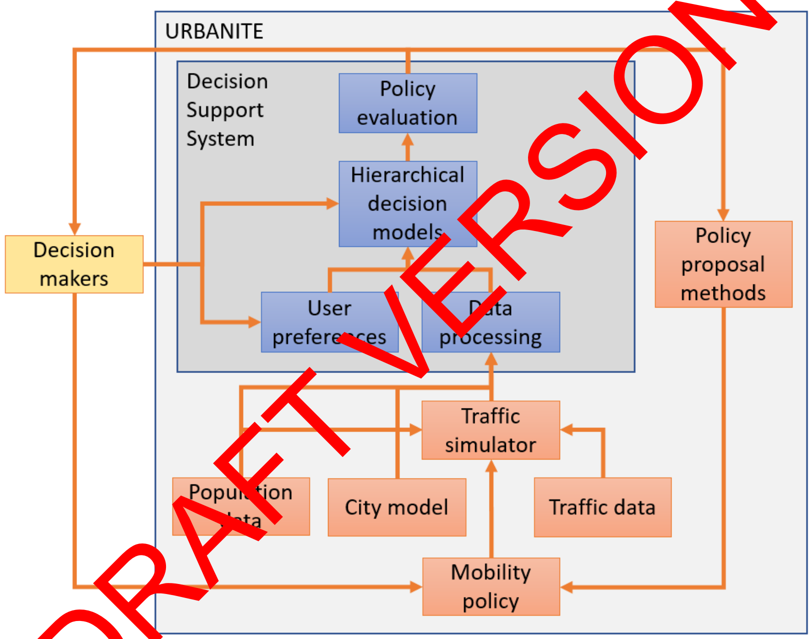

Figure 7 The architecture of the URBANITE system from the point of view of DSS.

The (basic) attributes of the DSS decision model are the KPIs that are measured by simulating the human mobility behaviour within the city, and the city properties. The objectives, including the overall objective are defined/selected by the city decision-makers. More precisely, the main inputs to DSS are:

1. The expert knowledge provided by the decision-makers. This knowledge is of key importance when building hierarchical decision models, as well as when defining user preferences.
2. Raw data including city models, population data, and evaluation results from the traffic simulator.
    a. Population data: the number of people in the urban area as well as their distribution between the districts.
    b. The city model: a map of roads, districts, public areas, etc.

c. Traffic simulation results: trip traces that include all the relevant data such as the (vehicle) positions, time, and pollution. These results are obtained by evaluating a mobility policy with the traffic simulator. To this end, the simulator processes the population data, the city model, and the past traffic data to emulate the characteristics of real-life traffic as much as possible.

The DSS has been incrementally developed. The initial version consists of a large set of attributes (i.e., KPIs and city properties) focused on target area, nearby area, and the entire city. It aims at evaluating the policies of all cities with a common decision model.

While analysing the city-specific objectives and attributes, it became clear that the cities have very different objectives as well as mobility simulation requirements. This resulted in disjunct simulations with differences in the used mobility modes etc. Consequently, the objectives that are calculated for a city, might not even be calculated for another city due to missing mobility modes etc. Therefore, we decided to create city-specific decision models as well as attributes. Both versions of DSS are described in the following sections.

3.1 Overview of the DSS approach

DSS evaluates mobility policies, i.e., for each policy it produces one or a limited set of objectives that are easily interpretable and handled by the experts. The traffic simulator already provides the baseline mobility policy evaluation; however, this evaluation is very difficult to process by experts due to a large amount of data since it consists of traces of all the trips within the city. To facilitate the task of decision-makers, the DSS aggregates the evaluation data into meaningful high-level attributes and objectives, which enables efficient and effective decision-making.

The DSS's main component is the hierarchical decision model. This model is defined by the experts/decision-makers based on their expert knowledge. It starts with the attributes, provided by the traffic simulator, and calculated from the city model, and iteratively combines semantically similar attributes into higher-level attributes and objectives until only one objective remains. This results in a tree structure in which the root represents the final evaluation of the policy (see the example in Section 2.2).

The model does not require the decision-maker to always use the final evaluation during the decision-making process. In some cases, it is more appropriate to use several objectives (e.g., pollution and congestion) to compare the policies in all the aspects that the decision-makers are interested in. In this case, the selected objectives are inner nodes of the tree structure and the process is multi-objective decision making (see Section 2.1).

The creation of the hierarchical decision model starts with the specifications of the user preferences. This step requires the involvement of the experts/decision-makers. When creating the decision model, these preferences are used to weight the attributes within the tree structure. More precisely, when combining attributes into a higher-level attribute or objective, a utility function needs to be defined. The utility function specifies how each combination of lower-level attributes transforms into the higher-level attribute or objective. This is a preference-based process and typically involves combining qualitative attributes of various types.

Note that the hierarchical decision models are not able to directly handle the city model data or the raw data obtained from the traffic simulator. Therefore, the data need to be preprocessed and, if appropriate, aggregated in order to obtain DSS attributes. For example, if the city pollution is required as an input to the hierarchical decision model, it has to be calculated from all the trips within the city.

Finally, policy evaluation has to be executed. This is done by applying the preprocessed traffic simulator data and city model data within the hierarchical decision model. The resulting values of the objectives, selected based on user preferences, are then sent to decision-makers or policy proposal methods (see Figure 7). The hierarchical decision models, including their definition and execution, were implemented with DEXi (see its description is Section 2.2).

3.2 Initial version of DSS

The initial hierarchical decision model is shown in Figure 8. It was developed based on the initial version of user needs and preferences of the URBANITE use cases. More precisely, the model was developed based on mobility policies that include building new roads, closing parts of the city like squares, setting up new lines of public transport including ferries, and other potential modifications of the city mobility. For a policy evaluation, three areas within the city were identified as relevant:

1. Target area where the policy action is applied
2. Nearby area that surrounds the target area and which is directly influenced by the applied policy
3. The entire city

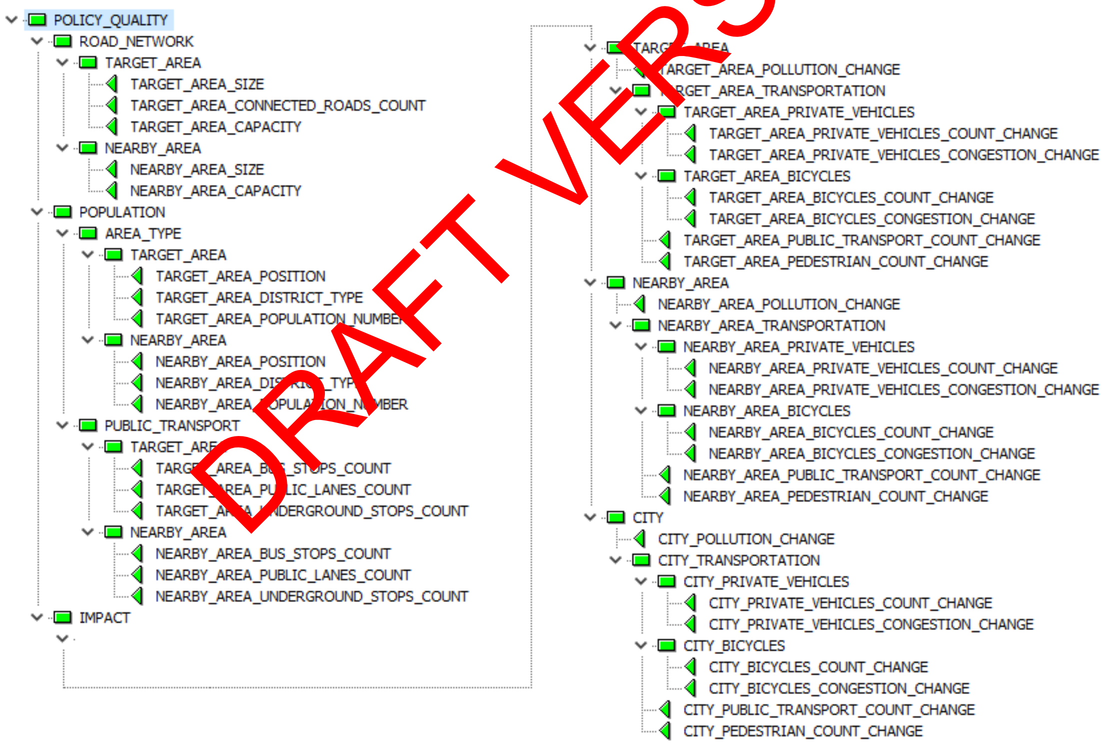

Figure 8 Initial hierarchical decision model.

The initial attributes, i.e., the initial KPIs and the initial city features were divided into three categories:

- Road network

These attributes measure the size of the city area where the policy action has a direct influence. They also consider the capacity of the affected roads taking into account both target and nearby areas.

- Population-related attributes including the type of the area and public transport data.

  Area type is defined with the position within the city (e.g., centre, periphery), the district type (e.g. residential, commercial), and the population number. Public transport counts the available bus and underground stops, and the lanes of public transport. All these attributes are measured in both target and nearby areas.

- Policy impact

  It measures the change with respect to the baseline scenario when no policy action is applied. The following aspects are considered:

  - Change in air pollution
  - Change in the number of used private vehicles
  - Change in the number of used bicycles
  - Change in the number of used public transport
  - Change in the number of pedestrians

  In addition, it also considers congestion change. All the attributes are measured in both target and nearby areas, as well as in the entire city.

This model is intended to be used for two purposes:

1. For comparing the effects of applying a policy with the baseline.
2. For comparing the effects of various policies between themselves.

Therefore, some attributes focus on comparison with baseline, while others focus on differences among various policies.

The utility functions (see the description in Section 3.1) were defined as follows. All the attributes in the inner nodes of the tree were defined as categorical from 1 to 5, which facilitated the utility function definition. The default scale defined the higher the better, except for the pollution where the lower-the-better scale was applied. For all nodes, similar utility functions were defined as the one shown in Figure 9.

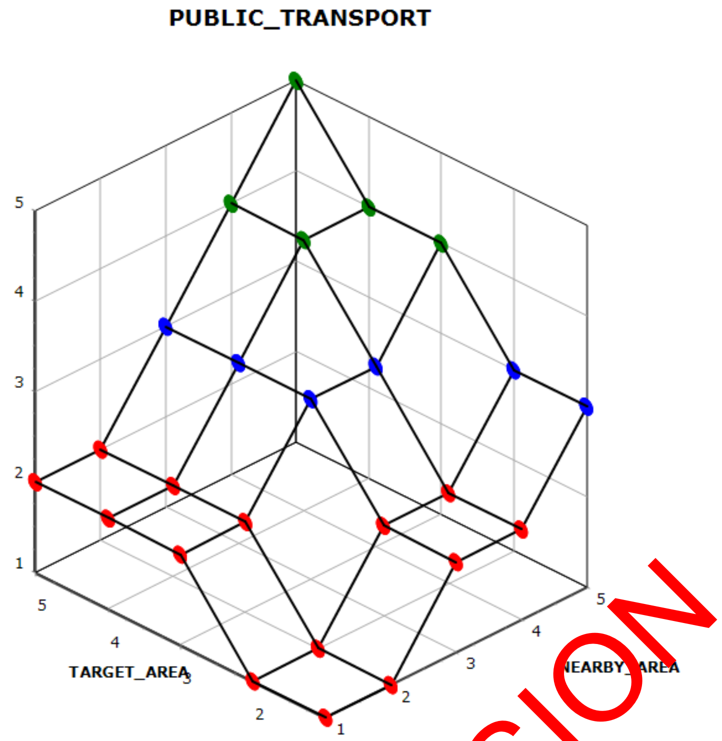

Figure 9 The utility function that aggregates attributes into the public transport objective.

3.3 City-specific DSS models and KPIs

After the initial decision model was designed, the use cases were redefined and additional specifications for the recommendation module were produced. To meet the updated specifications, the decision was made to create city-specific decision models. This is mostly due to the fact that it was clarified that the use cases have different needs which are also reflected in the use case KPIs. The resulting decision models are simpler and include only the aspects the pilot cities find useful.

Thus, the KPIs as well as the decision models are defined for each city based on the needs and interests of the use cases, as described in the following sections.

3.3.1 KPI calculation

KPIs represent metrics of specific policy outcomes the cities are interested in improving. To calculate expected values of these KPIs after some policy introduction, the policy change is simulated, and the simulation results are evaluated. Some of the KPIs are simple averages of some traffic-related values, however some of the KPIs are estimated using complex algorithmic methods and rely on other resources, such as the HBEFA for estimation of the air pollutant estimation. The KPIs are described and methods for calculation are detailed when appropriate, as the detailed explanation of all KPI calculation methods is out of scope of this document.

Most of the KPIs can also be visualised on the city map, before geospatial aggregation, as shown in Figure 10.

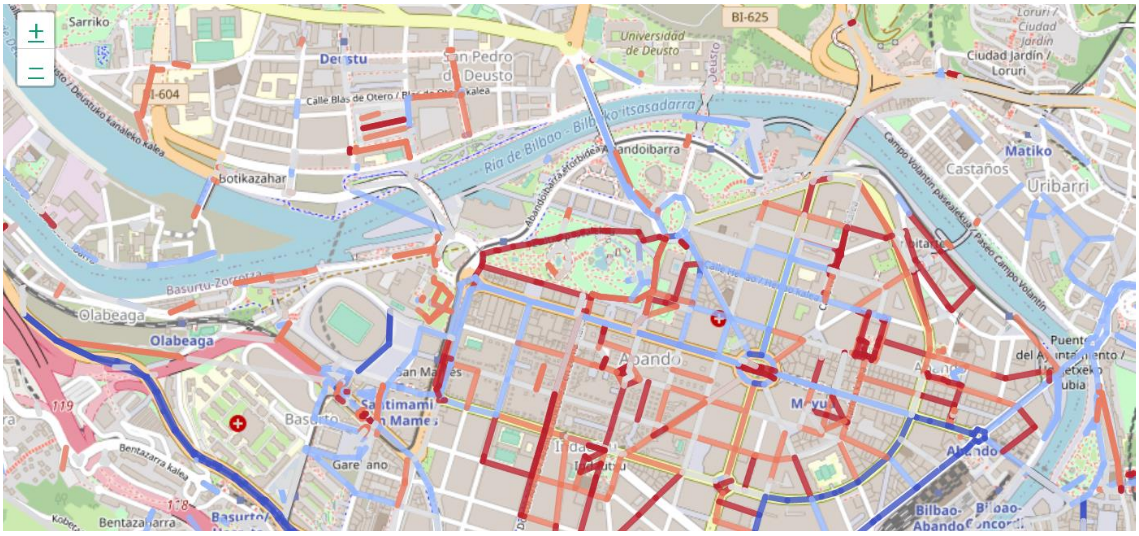

Figure 10 Visualisation of one of Bilbao KPIs on the map. The red lines show streets with high CO_2 emissions and the blue lines show streets with low CO_2 emissions

3.3.1.1 Use case: Amsterdam

The KPIs developed for the Amsterdam use case are the following:

- Bike congestions: measured on streets and bike lanes, segments with high utilisation of the bike infrastructure or segments with detected traffic jams, which are located and counted.
- Bike intensity: bike intensity is measured as a share of the maximum capacity of the streets and bike lanes. The numbers are aggregated geospatially and represent the amount of city bike infrastructure utilisation.
- Bike safety: a novel bike safety index was designed based on the number of cyclists and micro mobility users sharing lanes with cars and heavy vehicles.
- Bikeability: a novel bikeability index was designed based on the Munich bikeability index [4]. It includes a measure of bike infrastructure and street speed limits.

3.3.1.2 Use case: Bilbao

- Entry capacity to the central part during peaks: the maximum number of vehicles that can enter the central part of Bilbao without causing congestions. The central part is defined as the area surrounding the Moyúa square.
- Pedestrian travel time: measured as the average pedestrian trip time.
- Number of bike trips: the daily number of bike trips made.
- Share of trips by mode: the percentages of bike trips, bus trips and car trips of all trips. A trip is the part of a travel plan between two activities, made using one of the travel modes.
- Emitted air pollutants (NOx, PM10, CO2): daily air pollutant emissions based on the simulated vehicle movements and the HBEFA [5] database of emission factors.
- Acoustic pollution: based on the simulated vehicle movements and certain aspects of the roads, acoustic pollution is estimated using a simplified model based on the CNOSSOS-EU [6].

3.3.1.3 Use case: Helsinki

- Congestions and bottlenecks: segments with high utilisation and segments with detected traffic jams are located and counted.

- Traffic flow in the harbour area: the harbour area generates a lot of traffic towards the highway and rest of the city. This KPI measures the number of vehicles passing the Jätkäsaari smart junction, which connects the harbour area with the rest of the city.
- Emitted air pollutants (NOx, PM10, CO2): same as for the Bilbao use case.
- Acoustic pollution: same as for the Bilbao use case.

3.3.1.4 Use case: Messina

- Public transport usage: number of daily users of public transport.
- Average speed of buses and trams: daily average speed of all public transport vehicles.
- Number of bike trips: daily number of bike trips made.
- Share of trips by mode: same as for the Bilbao use case.
- Congestions and bottlenecks: same as for the Helsinki use case.

3.3.2 Multi-attribute decision models

The city-specific multi-attribute decision models are described in this section. Each hierarchical decision model is shown and the rationale for its design is provided.

3.3.2.1 Use case: Amsterdam

Amsterdam is currently one of the cities with most bicycle users. This causes some specific problems for the city, e.g., bike congestions and the need for improved cyclist safety. The decision model is split into local and city-wide effects, further split into traffic and bike infrastructure. The base attributes are the KPIs described in Section 3.3.1. Figure 11 shows Amsterdam's hierarchical decision model.

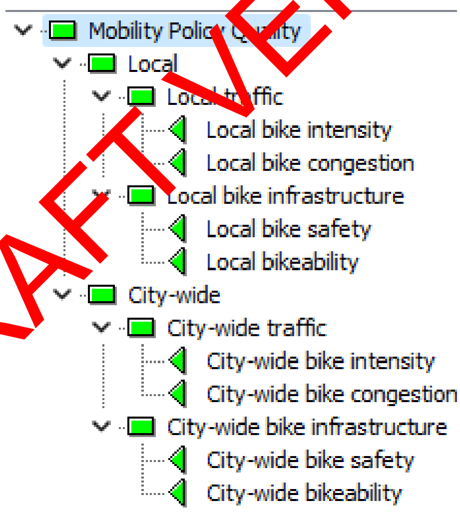

Figure 11 The hierarchical decision model developed for the Amsterdam use case. According to the needs of the city of Amsterdam, it is designed to support decision making about bicycle traffic.

3.3.2.2 Use case: Bilbao

The Bilbao use case is focused on improving the quality of life in the city and the decision model is defined accordingly. On the top level, the local and city-wide aggregated attributes represent the effect of the proposed policies on the target area and the entire city. In each area pollution and traffic are considered. Pollution includes air pollutant emission, specifically NOx, PM10 and CO2, as well as the acoustic noise pollution. The traffic attribute is also composed of attributes related to quality of life, entry capacity to centre, pedestrian travel time, and daily number of bike travels. Figure 12 shows Bilbao's hierarchical decision model.

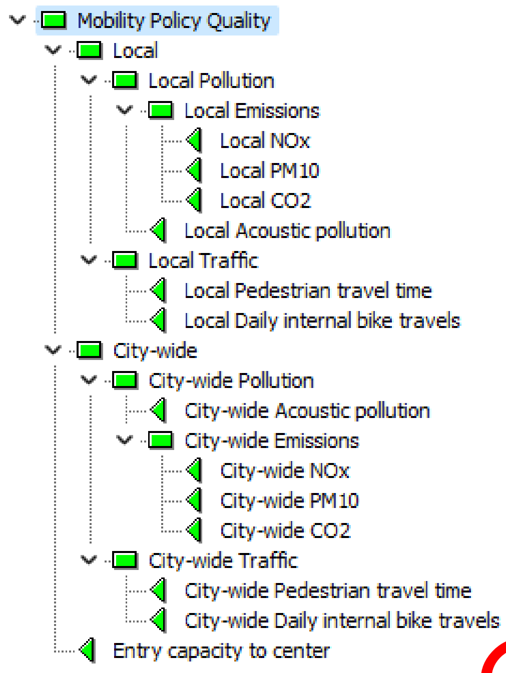

Figure 12 The hierarchical decision model developed for the Bilbao use case. Since Bilbao is interested in improving the quality of life in the city, the selected attributes include air pollutant emission, acoustic pollution, and quality of life related traffic attributes.

3.3.2.3 Use case: Helsinki

The main interest of the Helsinki use case is the large amount of traffic arriving via ferries to the harbour in Jätkäsaari, which has only a single point of transfer to the mainland. The road connecting the harbour with the mainland is often congested, which causes a lot of air pollution and acoustic noise. On the top level we include local attribute values, city-wide values, and the traffic flow in the harbour area. Figure 13 shows Helsinki's hierarchical decision model.

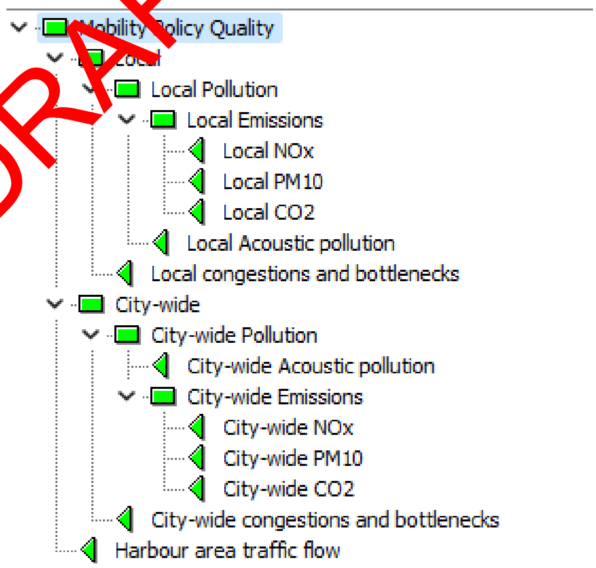

Figure 13 The hierarchical decision model developed for the Helsinki use case. It differs from others by having the harbour area traffic flow on the top level according to the city's interests.

3.3.2.4 *Use case: Messina*

The Messina use case is focused on improving public transport while lowering the number of trips made using personal cars. The decision model is split on the top level into local and city-wide attributes. The public transport attribute is composed of the number of public transport trips, average speed of the public transport and the number of bike trips. The share of trips attribute represents shares of trips made using different travel modes, where larger shares are better for share of bike trips and public transport trips, but a smaller share of car trips is better. The Messina's hierarchical decision model is shown in Figure 14.

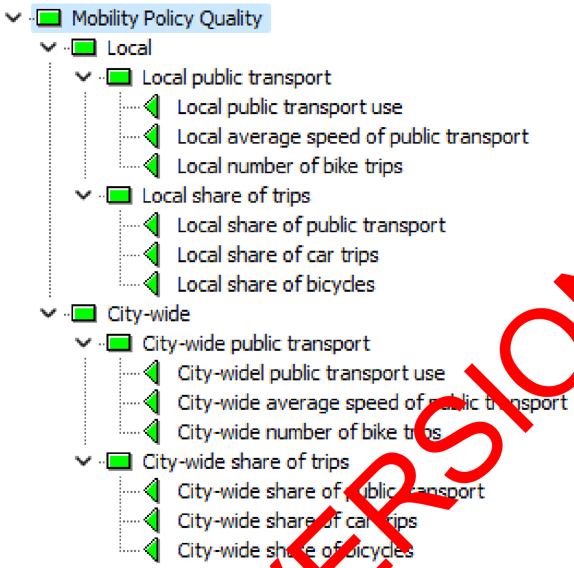

Figure 14 The hierarchical decision model developed for the Messina use case. Since Messina is interested in improving public transport, the decision model includes public transport related attributes and shares of use of different transportation modes.

3.4 Policy effect modelling and policy proposal

Policy proposal upgrades the approach for evaluating mobility policies, described in Sections 3.1-3.3, and shown in Figure 7, with the approaches for supporting decision-makers in the selection of the best-suited policies. To this end, two main functionalities are provided:

1. Presentation of the evaluated policies to decision-makers with the goal of enabling decision-makers to efficiently select the best mobility policies.
2. Automatic discovery of the best mobility policies with respect to the given objectives/preferences.

3.4.1 Mobility policy presentation

The presentation of the mobility policies with respect to the objectives enables decision-makers to select the best mobility policies. We developed the capabilities for policy representation for the Bilbao use case. For this use case, 14 simulations were executed by applying different policies. These simulations represent two different scenarios:

- Baseline scenario of the current city situation
- Modified scenario representing closure of the Moyua square for private traffic

The first two simulations refer to the aforementioned scenarios respectively, while the remaining are variations of the second scenario. These variations are related to changes in the number of cyclists from 1.500 to 19.000 while the number of inhabitants varies from 2.000 to 20.000. The change in number of cyclists does not represent a specific policy but shows what would happen if in conjunction with the applied policy also the number of cyclists changes due to changed behaviour or additional policy. Selected GUIs are presented in Figures 15-16 that show the selected objectives. These figures show that by changing the number of cyclists in combination with closure of the Moyua square for private traffic, the level of CO2 emissions can be decreased. Thus, the decision-makers can select the preferred policies with respect to the target emission decrease.

This is the first version of the GUIs. In the future development, we will extend the functionalities to all use cases, as well as provide support for multi-objective decision making.

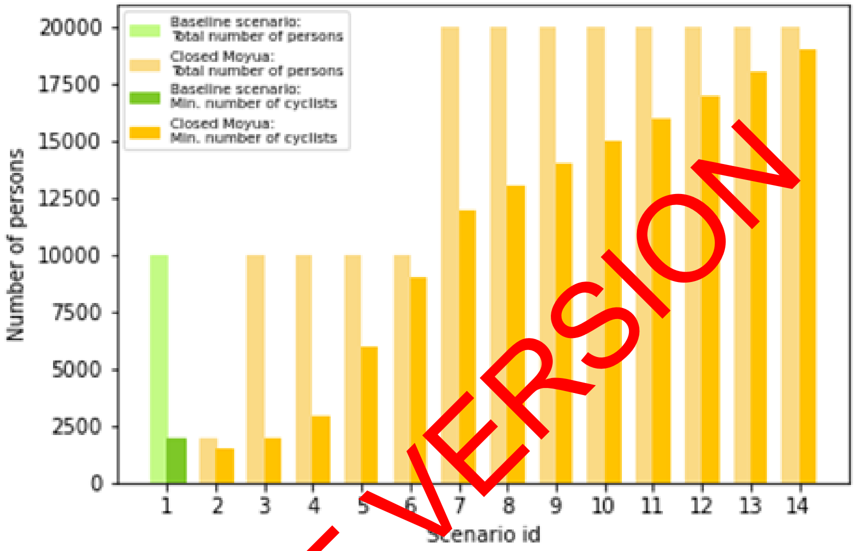

Figure 15 Results of the scenarios for closure of the Moyua square where green marks the baseline scenario and orange the other scenario and all its variations.

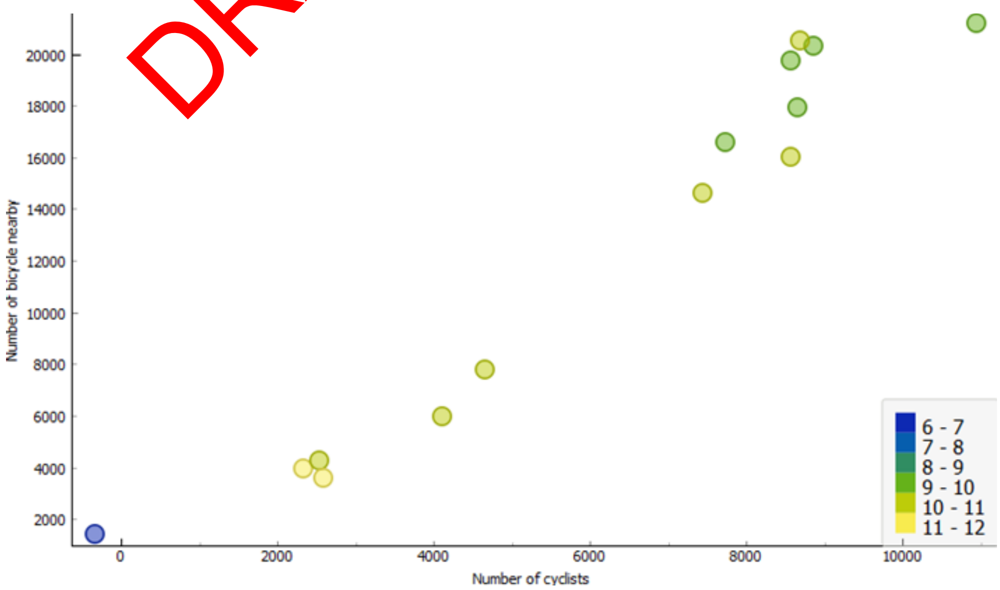

Figure 16 The results of the applied policy with respect to two selected objectives. The x-axis represents the number of cyclists nearby the square and the y-axis represents the number of cyclists in the centre. The different colours denote the amount of CO2 emissions as a target variable where an orange circle marks the baseline scenario.

3.4.2 Machine learning approach for policy proposal

The recommendation system applies machine learning (ML) models to automatically propose mobility policies. ML models use pattern-recognition approaches for finding and proposing the best mobility policy. To this end, the effect of the proposed policies is modelled based on the qualitative results obtained from the DSS. Since multiple cities need to be taken into consideration, each having their own set of attributes, we consider aggregation of the qualitative results to be representative of the effect that the mobility policies have per each city independently. The goal of the automatic policy proposal is to enable decision-makers to define the desired scenario, i.e., the desired change in the city, while the policy proposal module automatically finds and outputs a policy that best correlates the desired changes with predicted results.

A policy proposal module has to be robust enough to allow multiple objectives to be satisfied as is the user's choice. In order to do that, we use the advantage of having discrete attributes in our hierarchical decision model. Our recommendation system takes a subset of elements that contain the same value for the objective(s) that needs to be optimised. For instance, if it is the decision-makers wish to improve the local acoustic pollution in the city of Helsinki, the system aggregates all the data which contains the categorical value of "good" for noise pollution. From the aggregated data, various attribute distributions are recorded and later sampled in order to create a synthetic population. Doing so, the newly created data is closely similar to the original data that was provided to the DSS, thus ensuring the creation of meaningful policies.

In addition to the distribution-based approach described above, machine-learning methods are used in the policy proposal module. In this case, the recommendation system learns from the results of the microscopic traffic simulations. The main idea is to use one simulation run as one training example. The training data consists of several groups of parameters that are related to the input and output of the simulation, and the key performance indicators (KPIs).

The machine-learning approach is formulated as follows. The ML models should predict the mobility measures (e.g., close a street) based on the preferred objectives (e.g., pollution reduction). An example of the ML problem is as follows:

- Features:
  - CO2, NOx, PM10, Bikeability, Bike safety, share of car trips, share of bike trips, Share of PT trips
  - All of them are discretized into 5 intervals: -15%, -5%, 0%, 5%, 15%. These intervals represent the changes compared to the baseline scenario where no mobility measure is applied, i.e., the current status within the city.
  - Features are objectives in the hierarchical decision models (see Sections 3.2-3.3) that are calculated from the results of the simulations.
- Target variables:
  - Road/square closure start: discretized into half an hour interval (1, …, 48)
  - Road/square closure duration: discretized into 15-minutes intervals (1, …, 16)
  - Closure size: the length of closed roads in km
  - Target variables are decisions/measures that are applied in the city and represent the inputs to the simulations.

The created ML models are used as follows. The decision-makers should define the preferred values of the features, i.e., the required amount of reduced CO_2, NO_x, PM10, etc. The ML models then predict which mobility measures should be applied to obtain the given effect, e.g., a street of X km long has to be closed for Y minutes after Z time.

The ML problem is multitarget since several target variables need to be predicted simultaneously. The following ML algorithms that support multitarget prediction are suitable for this problem.

Linear regression

Linear regression weights and sums the input features to calculate the target variables. Thus, the relationships between inputs and outputs are modelled with linear predictor functions.

Nearest neighbours

Nearest neighbours find the nearest (closest) neighbours in the feature space by the means of Euclidean distance and averages the target variables of these neighbours to predict the target variables of the observed instance.

Decision tree

The decision tree algorithm implements a tree structure, where each internal node represents a condition for a feature, each branch represents satisfaction of the node condition, and each terminal node determines the target values assigned to the instances that met the conditions of the internal nodes on the path from the root node to the terminal node.

Random forest

Random forest applies a set of decision trees, where each decision tree is built only on a random subset of data. In addition, it performs random feature selection at node partitioning, i.e., when partitioning a node, a subset of features is randomly selected and only those features are considered during the partitioning. The final prediction is determined by averaging the results of the underlying decision trees.

Since heterogeneous mobility measures are envisioned, we will hierarchically combine the ML models into two levels. At the top level, the model will predict which measure(s) should be applied. When a measure will be selected, the bottom-level model will specify the details of this measure. For example, at the top level, we will predict whether a street should be closed or not. If closure is selected, the bottom level will specify the start and duration of the closure.

3.4.3 Evolutionary algorithms for policy proposal

Evolutionary algorithms (EAs) are being developed as complementary procedures to ML models for the proposal of the mobility policies. EAs implement search procedures that do not need training examples given in advance but require to define the objectives that need to be optimised. The overall set of objectives for our problem is presented in Sections 3.2-3.3, while the actual objectives should be selected by the decision-makers based on the current needs.

Evolutionary algorithms

Evolutionary algorithms (EA) are search algorithms that mimic natural selection and natural genetics [7]. Their main principles are solution representation, solution evaluation, solution improvement in several iterations called generations, selection operator and genetic operators (crossover and mutation) and stopping condition. EA starts with a set of random solutions called population. These solutions are then improved through several generations by selecting some of current solutions, modifying them with crossover and mutation, evaluating them, and adding them to the population, i.e., replacing existing solutions in the population, if the new solutions are better than existing ones. This procedure continues until stopping conditions are met, e.g., maximum number of generations, maximum execution time, or convergence.

Solution representation

Each solution represents a mobility policy that consists of a set of mobility measures. The policy is encoded as a vector of values that enable or disable, and further specify all possible mobility measures. Figure 17 presents an example of solution representation.

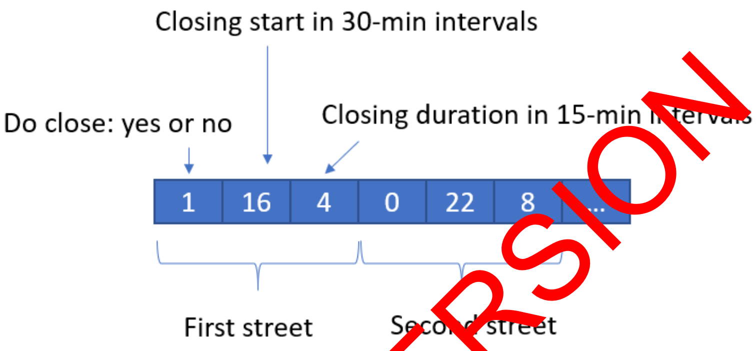

Figure 17 An example of the representation of mobility policies in EAs.

Crossover and mutation operators

Crossover and mutation operators change the solutions to produce new solutions. Crossover takes two solutions, randomly selects the crossover points, and exchanges the data of these solutions between every second pair of crossover points. Mutation randomly changes a randomly selected part of a solution. Both operators are executed with some probability, so they are not executed at every step. The crossover is schematically presented in Figure 18.

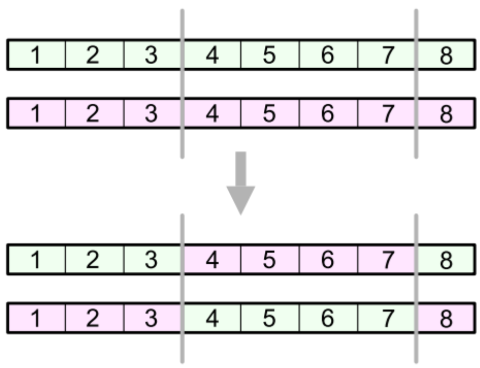

Figure 18 Schematic presentation of crossover with two crossover points.

Solution evaluation

When evaluating mobility policies, several objectives are considered simultaneously, such as CO2, NOx, PM10, Bikeability, Bike safety, share of car trips, share of bike trips, Share of PT trips. The decision-maker can combine these objectives into higher-level objectives based on the hierarchical decision models (see Sections 3.2-3.3). Note that having less higher-level objectives facilitates the EA search and enables to find better solutions in a shorter time. Nevertheless, multiple objectives are efficiently handled by EAs since dedicated EAs are specifically designed for multi-objective decision problems (e.g., NSGA-II [8]).

Note that the development of the policy proposal module is ongoing, and it will be improved till the conclusion of Task 4.2 (in Month 30). Thus, additional improvements of the module are currently under implementation and are expected to enable finding good policies with several interacting policy actions and compared with several objectives.

4 Delivery and usage

4.1 Decision Support System

4.1.1 Installation instructions

DEXi[1] is an interactive computer program for multi-attribute decision making, supporting complex decision-making tasks. A multi-attribute model is a hierarchical structure that represents the decomposition of the decision problem into subproblems.

4.1.2 User Manual

The user manual is available in https://kt.ijs.si/MarkoBohanec/pub/DEXiManual505.pdf.

4.1.3 Licensing information

The license terms for the software are under discussion among the consortium. AGPLv3[2] are being considered.

4.1.4 Download

You can download the program from the web page: https://kt.ijs.si/MarkoBohanec/DEXi/setup/DEXi505en_setup.exe.

4.1.5 Application in the context of URBANITE use cases

Data and configuration files for the analysis and application to URBANITE, described on D4.3, can be found on https://git.code.tecnalia.com/urbanite/private/wp4-algorithms-and-simulation/models_and_working_files/-/blob/main/Policy%20Decision%20Model/Policy_Decision_Model.7z, on the GitLab maintained by Tecnalia[3].

---

[1] https://kt.ijs.si/MarkoBohanec/dexi.html

[2] https://www.gnu.org/licenses/agpl-3.0.en.html

[3] https://git.code.tecnalia.comNgx-admin - most popular admin dashboard on Angular 9+ and Nebular. (akveo.github.io)

This directory includes data and configuration files, that defines the URBANITE Policy Decision Model:

- DEXi city-specific models (Bilbao, Helsinki, Amsterdam, and Messina)
- Code for the calculation of the different KPIs and analysis: bikeability_index, events, network
- Experiments data and results

5 Conclusions

This deliverable presents the developed URBANITE Decision Support System. DSS enables decision-makers to select the relevant objectives, evaluate mobility policies, and obtain better mobility policies with the use of the recommendation system. This deliverable describes the methodological background of the multi-attribute and multi-objective decision making, and the DSS approach that was developed based on this background. Two versions of DSS are described: the initial one and the city-specific one. Both are based on the use case requirements. Finally, it describes the approach for policy effect modelling and policy proposal.

As this is an intermediate deliverable of Task T4.2, it does not describe the final version of the recommendation system, but it gives an overview of the status of its current development. The final version of the recommendation system will be available in Month 30 and described in deliverable D4.6.

The future work mainly consists of developing the final version of the recommendation system. In addition, the use cases might be further redefined although they were already defined. This would require modifications of DSS, although significant modifications are not foreseen, since the module is very flexible and enables to redefine the decision models on the fly. However, if new KPIs are defined, the data pre-processing module will need to be upgraded.

[3] Angular
/urbanite/private/wp3-data-management/storage/dataStorage

6 References

[1] K. Deb, «Multi-Objective Optimization using Evolutionary Algorithms: An Introduction,» 2011.

[2] M. Bohanec, «DEXi: Program for Multi-Attribute Decision Making, User's Manual, Version 5.04,» IJS Report DP-13100, Jožef Stefan Institute, Ljubljana, 2020.

[3] Dovgan, E., Smerkol, M., Sulajkovska, M., Gams, M., «Supporting decision-making in the urban mobility policy making,» 24th international multiconference Information Society (IS), Ljubljana, 2021.

[4] Schmid-Querg, J., Keler, A., & Grigoropoulos, G., «The Munich Bikeability Index: A practical approach for measuring urban bikeability. Sustainability, 13(1), 428,» 2021.

[5] Notter, B., Keller, M., Althaus, H. J., Cox, B., Knörr, W., Heidt, C., et al., «HBEFA 4.1. INFRAS,» 2019.

[6] Kephalopoulos S, Paviotti M, Anfosso-Lédée F., «Common Noise Assessment Methods in Europe (CNOSSOS-EU),» Publications Office of the European Union, Luxemburg, 2012.

[7] Man, K. F., Tang, K. S., Kwong, S.;, «Genetic Algorithms,» Springer, London, 1999.

[8] Deb, K., Pratap, A., Agrawal, S., Meyarivan, T., «A fast and elitist multiobjective genetic algorithm: NSGA-II.,» 2002.

[9] Demšar, J., Curk, T., Erjavec, A., Gorup, Č., Hočevar, T., Milutinovič, M., et al., «Orange: data mining toolbox in Python. the Journal of machine Learning research, 14(1), 2349-2353.,» 2013.